Department of Computer Science and Engineering
Institute of Technical Education & Research, SOA, Deemed to be University

PRACTICE ASSIGNMENT
UNIX Systems Programming (CSE 3041)

Q1. Create a chain of n-process. Pass n on the command line. Each process terminates only after child of process finished its execution. Before process terminate print its process id and parent process id.

Q2. Write a program that creates a child process to run ls -l.

Q3. Write a program to create a zombie child process. Check its status using *ps* command .

Q4. Write a program to output current working directory.

Q5. Write a program which will copy a file from one directory to another directory. Pass source of file and destination of file on command line. Your code will check for the correct number of arguments. If it is not correct, then an error message will be produced and the code will exit. The parent will wait for the child to finish. Exa.

$mycp source_path_file1 destination_path_file2

Q6. If you try to print a message after the exec* call, does it print it? Why? Why not?

Q7. How many times **iter** is printed and how many processes are create by following code including program process.

```
Int main()
{
 int pid,pid2;
pid=fork();
if(pid)
{
        pid2=fork();
        printf("iter");
}
else
{
      printf("iter");
     pid2=fork();
}
return 0;
}
```

Q8. Modify above program so that you can verify how many processes created by above code.